

MIT App Inventor

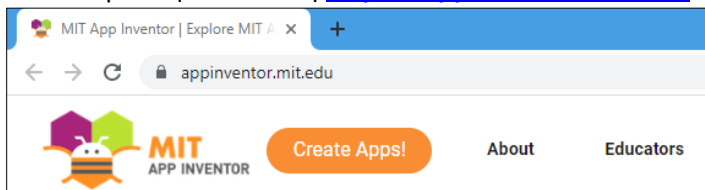
(<https://appinventor.mit.edu/>)

Το MIT App Inventor είναι ένα προγραμματιστικό περιβάλλον με το οποίο μπορούμε να φτιάξουμε εφαρμογές για συσκευές που λειτουργούν με android ή iOS (κινητά τηλέφωνα, τάμπλετ).

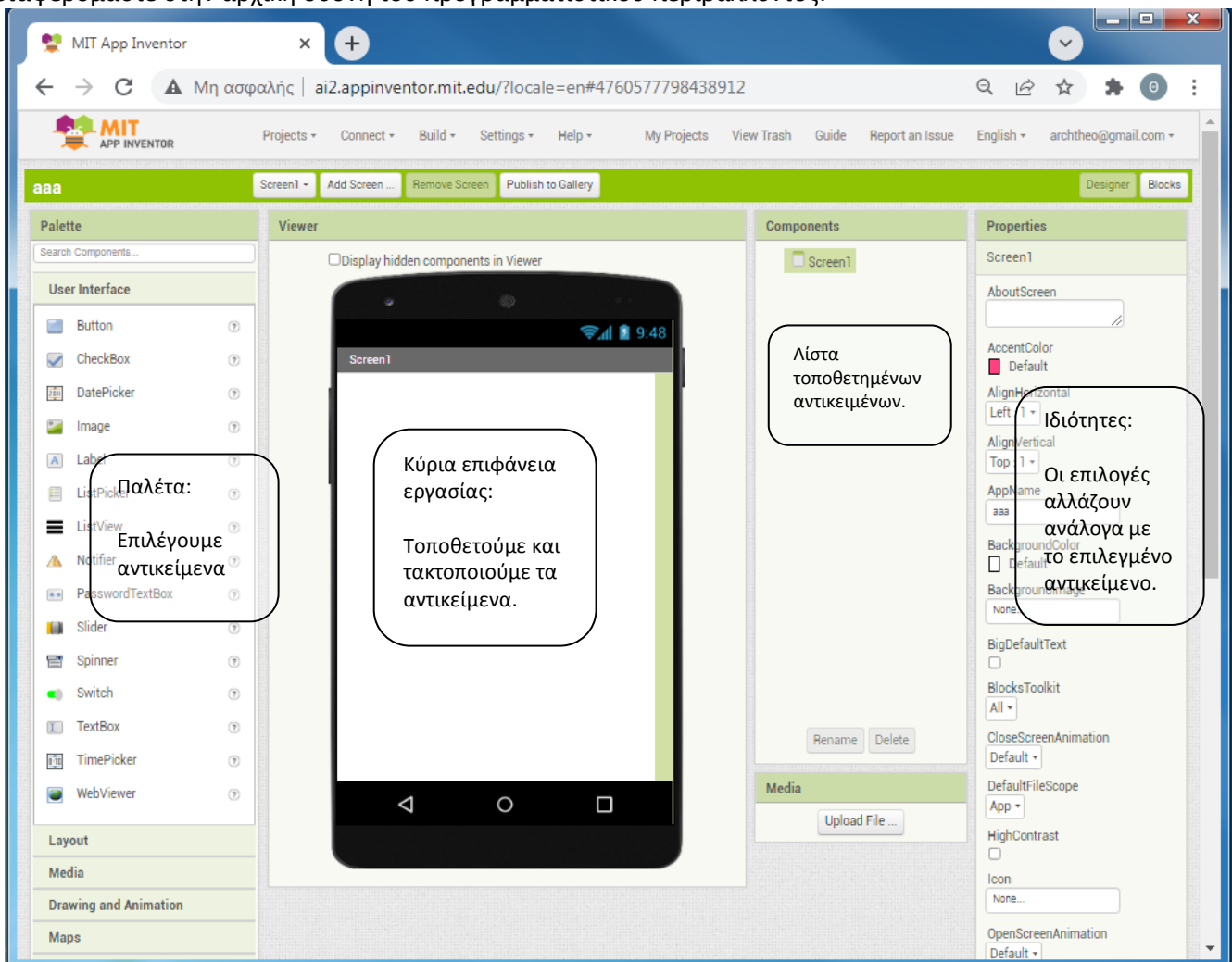
Προγραμματίζουμε στον υπολογιστή (το περιβάλλον μοιάζει πολύ με το scratch) και μεταφέρουμε την εφαρμογή και την εκτελούμε στο κινητό!

Δεν χρειάζεται εγκατάσταση κάποιου λογισμικού. Όλα γίνονται απευθείας μέσα από το περιβάλλον του φυλλομετρητή. Προϋπόθεση είναι να έχουμε λογαριασμό στο google.

Μπαίνουμε στη διεύθυνση <https://appinventor.mit.edu/>



Πατάμε το κουμπί Create Apps! και συνδεόμαστε με τους κωδικούς που έχουμε για το google. Μεταφερόμαστε στην αρχική οθόνη του προγραμματιστικού περιβάλλοντος:

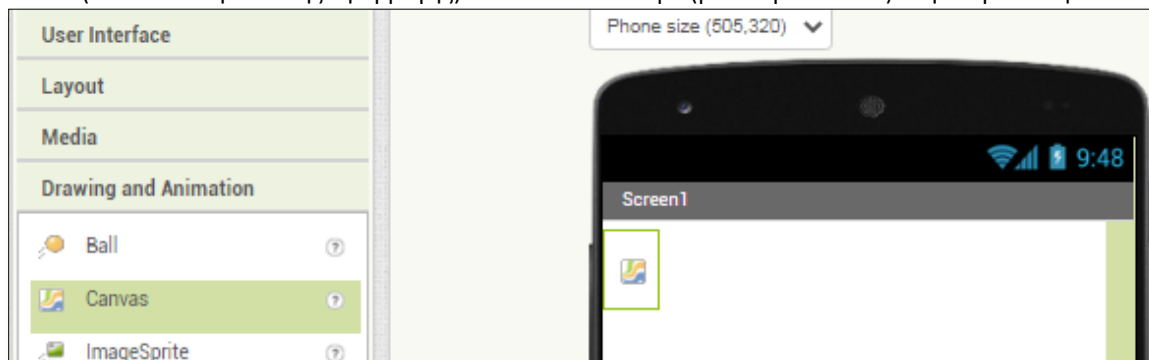


Πάμε να φτιάξουμε την πρώτη μας εφαρμογή (Μπάλα που αναπηδά):

Projects, Start New project. Της δίνουμε το όνομα **BallBounce** (δεν επιτρέπονται κενά)

Σχεδιασμός:

Από την παλέτα επιλέγουμε την ομάδα αντικειμένων **Drawing and Animation** και από αυτήν επιλέγουμε το αντικείμενο **Canvas** (θα είναι το φόντο της εφαρμογής) και το τοποθετούμε (με «σύρε κι άσε») στην κύρια επιφάνεια εργασίας:

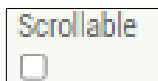


Ρυθμίζουμε την οθόνη της εφαρμογής μας να μην «κυλάει» όταν κάποιο αντικείμενο φτάνει στα όριά της, γιατί θέλουμε η μπάλα να αναπηδάει όταν χτυπάει στα όρια της οθόνης:

Κάνουμε κλικ στην κύρια επιφάνεια εργασίας.

Στη δεξιά στήλη (ιδιότητες), επάνω πρέπει να γράφει Screen1.

Βρίσκουμε την ιδιότητα Scrollable και την αποεπιλέγουμε (αν δεν είναι ήδη αποεπιλεγμένη):

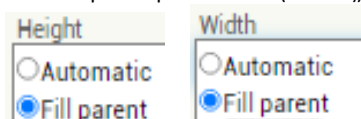


Τροποποιούμε τις διαστάσεις του αντικειμένου Canvas έτσι ώστε να γεμίζει την οθόνη του κινητού όταν εκτελείται η εφαρμογή (αφού είναι το φόντο):

Επιλέγουμε το αντικείμενο Canvas.

Στη στήλη των ιδιοτήτων εμφανίζεται το όνομά του.

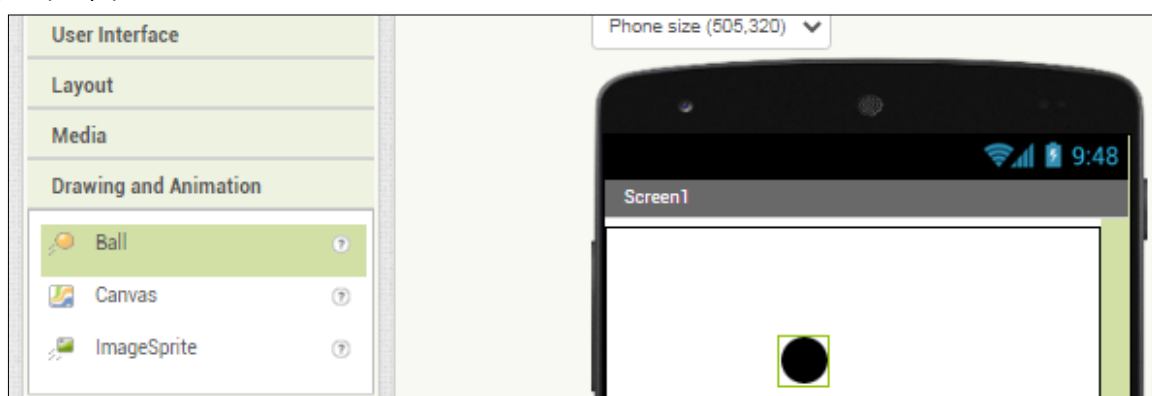
Βρίσκουμε κι επιλέγουμε την ιδιότητα Height (ύψος) και επιλέγουμε Fillparent. Το ίδιο κάνουμε και με το Width (πλάτος):



Προσθέτουμε την μπάλα:

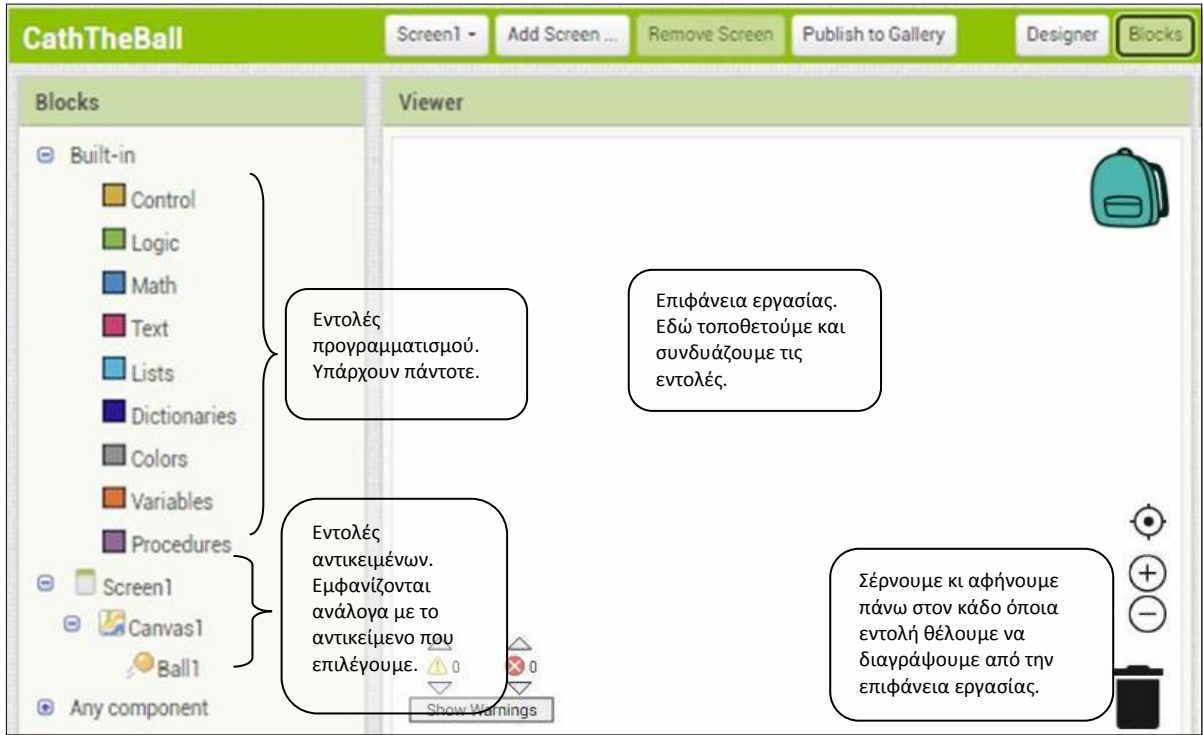
Από την παλέτα επιλέγουμε και πάλι την ομάδα Drawing and Animation και στη συνέχεια το αντικείμενο Ball και το

τοποθετούμε στην κύρια περιοχή. Στη συνέχεια το επιλέγουμε και από τις ιδιότητές του βρίσκουμε την ακτίνα του (Radius) και της δίνουμε την τιμή 15:



Ήρθε η ώρα του προγραμματισμού!

Από την οθόνη σχεδιασμού μεταφερόμαστε στην οθόνη προγραμματισμού κάνοντας κλικ στο κουμπί Blocks πάνω δεξιά:



Κάνουμε κλικ στο στοιχείο Ball1 και αυτομάτως εμφανίζονται όλες οι διαθέσιμες εντολές (blocks) για αυτό:



Επιλέγουμε την εντολή **whenBall1.Flung do**
(τι θα γίνει όταν «εκσφενδονίσουμε» την μπάλα):

```
when Ball1 .Flung
  x y speed heading xvel yvel
do
```

(Θα θέλουμε η μπάλα να κινηθεί προς την κατεύθυνση που την σπρώξαμε με το δάχτυλό μας και με ταχύτητα ανάλογη με την ταχύτητα με την οποία την σπρώξαμε.)

Ρυθμίζουμε την κατεύθυνση και την ταχύτητα της κίνησης:
Κάνουμε κλικ στο στοιχείο Ball1 και στη συνέχεια στην εντολή **setBall1.Heading to** και στην εντολή **setBall1.Speed to**:

```
set Ball1 . Heading to
set Ball1 . Speed to
```

και τις βάζουμε μέσα στη «δαγκάνα» της αρχικής εντολής (όταν τις τοποθετήσουμε σωστά «κουμπώνουν»):

```
when Ball1 .Flung
  x y speed heading xvel yvel
do
  set Ball1 . Heading to
  set Ball1 . Speed to
```

Ακουμπώντας τον δείκτη του ποντικιού πάνω στη λέξη speed (όχι κλικ, απλά αφήνουμε τον δείκτη για λίγο πάνω στη λέξη) βλέπουμε μια καινούρια εντολή, την **getspeed**:

```
when Ball1 .Flung
  x y speed heading xvel yvel
do
  set Ball1 . Heading to
  set Ball1 . Speed to
  set speed to
```

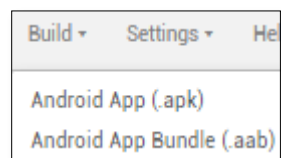
Την επιλέγουμε και την κουμπώνουμε στην αντίστοιχη εντολή.
Κάνουμε το ίδιο και την κατεύθυνση (heading).
Το μπλοκ των εντολών μας θα γίνει έτσι:

```
when Ball1 .Flung
  x y speed heading xvel yvel
do
  set Ball1 . Heading to get heading
  set Ball1 . Speed to get speed
```

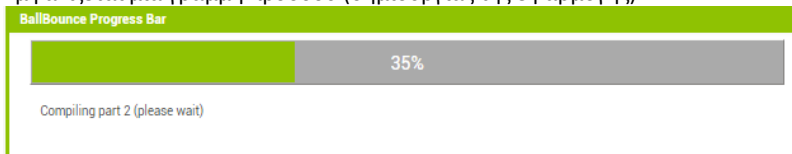
Αν θέλουμε να ακυρώσουμε (σβήσουμε) κάποια εντολή, τη σέρνουμε και την αφήνουμε στον κάδο κάτω δεξιά στην οθόνη.

Δοκιμάζουμε την εφαρμογή μας:

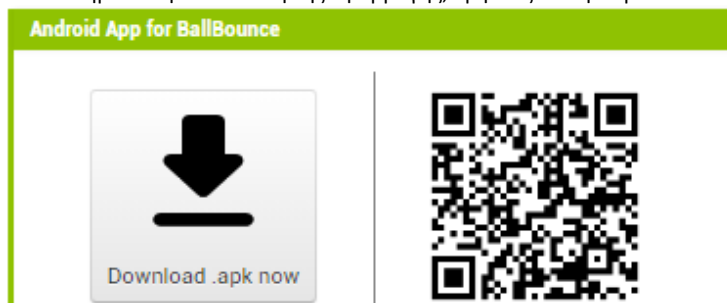
Επιλέγουμε Build, Android App (.apk)



Εμφανίζεται μια γραμμή προόδου (δημιουργίας της εφαρμογής):



Όταν ολοκληρωθεί η κατασκευή της εφαρμογής, εμφανίζεται η παρακάτω οθόνη:



Σαρώνουμε με το κινητό μας το QRCode και ακολουθούμε τις οδηγίες για την εγκατάστασή της.

Αν εμφανιστούν μηνύματα για επιβλαβές περιεχόμενο, επιλέγουμε ότι η εφαρμογή μας δεν είναι επικίνδυνη και προχωράμε στην εγκατάστασή της για να τη δοκιμάσουμε.

Τι διαπιστώνουμε;

Ότι η μπάλα φτάνει στα όρια της οθόνης και κολλάει...

Γιατί;

Επειδή δεν προγραμματίσαμε τι θα γίνεται όταν η μπάλα φτάνει στα όρια της οθόνης!

Συνεχίζουμε τον προγραμματισμό:

Επιλέγουμε το στοιχείο Ball1 και ακολούθως την εντολή **whenBall1.EdgeReached do** (τι θα γίνει όταν η μπάλα ακουμπήσει στα όρια της οθόνης):

```
when Ball1 .EdgeReached
  edge
do
```

Επιλέγουμε Ball1 και την εντολή **callBall1. Bounceedge do** και την κουμπώνουμε στην προηγούμενη. Ακολούθως αφήνουμε τον δείκτη του ποντικιού πάνω από τη λέξη edge για να εμφανιστεί η εντολή **get edge** την οποία κουμπώνουμε στην προηγούμενη εντολή.

Όλο το μπλοκ εντολών θα φαίνεται έτσι:

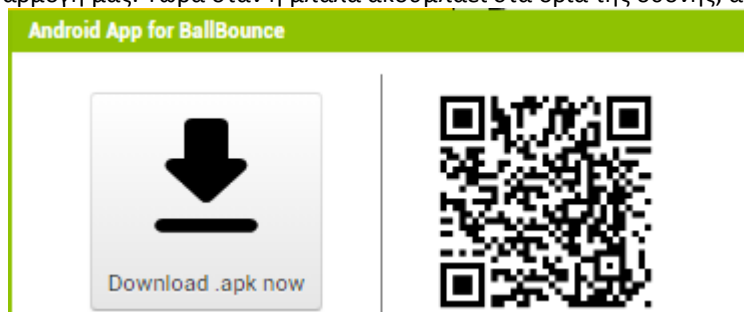
```
when Ball1 .EdgeReached
  edge
do call Ball1 .Bounce
  edge get edge
```

Το σύνολο των εντολών της εφαρμογής μας θα φαίνεται έτσι:

```
when Ball1 .Flung
  x y speed heading xvel yvel
do set Ball1 .Heading to get heading
  set Ball1 .Speed to get speed

when Ball1 .EdgeReached
  edge
do call Ball1 .Bounce
  edge get edge
```

Δοκιμάζουμε και πάλι την εφαρμογή μας. Τώρα όταν η μπάλα ακουμπάει στα όρια της οθόνης, αναπηδάει.

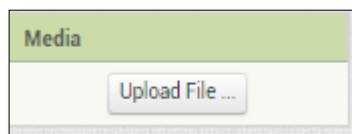


Δεύτερη εφαρμογή: MagicTrick (Καπέλο με κουνέλι)

Για την κατασκευή της εφαρμογής θα χρειαστούν: Μία εικόνα ενός άδειου καπέλου, μία εικόνα του ίδιου καπέλου με ένα κουνέλι να βγαίνει μέσα από αυτό και ένα αρχείο ήχου ως εφέ όταν θα εμφανίζεται το κουνέλι.

Σχεδιασμός:

Κάτω από τη στήλη των στοιχείων (Components) υπάρχουν τα «μέσα» (Media),



δηλαδή τα αρχεία εικόνων, ήχων, βίντεο που έχουμε «φορτώσει» για να χρησιμοποιήσουμε στην εφαρμογή μας.

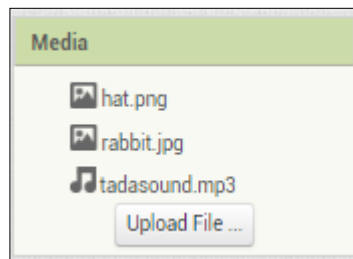
Φορτώνουμε τα τρία παραπάνω αρχεία (καπέλο, καπέλο με κουνέλι, ήχο). Αν δεν έχουμε έτοιμα, ψάχνουμε και βρίσκουμε στο διαδίκτυο.

Κάνουμε κλικ στο κουμπί «Upload File...» και στο ομώνυμο παράθυρο διαλόγου που θα εμφανιστεί κάνουμε κλικ στο κουμπί «Επιλογή αρχείου».

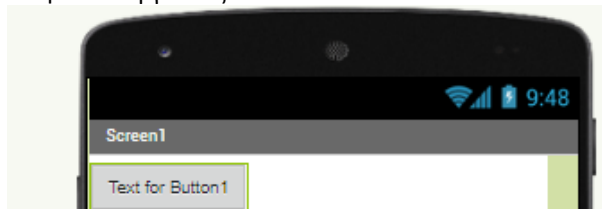
Στη συνέχεια βρίσκουμε κι επιλέγουμε το αρχείο που θέλουμε. Επιστρέφουμε στο παράθυρο διαλόγου Upload File... όπου τώρα φαίνεται και το όνομα του αρχείου που επιλέξαμε. Πατάμε το κουμπί OK και στη στήλη Media βλέπουμε το αρχείο που επιλέξαμε.

Επαναλαμβάνουμε όσες φορές χρειάζεται για να φορτώσουμε όλα τα αρχεία που χρειαζόμαστε.

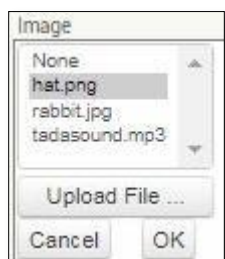
Στο τέλος η στήλη Media θα φαίνεται κάπως έτσι:



Από την παλέτα επιλέγουμε την ομάδα **User Interface** και από αυτήν το αντικείμενο **Button** και το τοποθετούμε στην κύρια επιφάνεια εργασίας:



Το επιλέγουμε και στις ιδιότητές του βρίσκουμε το Image, κάνουμε κλικ στην τιμή None... και από το μενού που εμφανίζεται επιλέγουμε το αρχείο με το καπέλο και πατάμε το κουμπί OK:

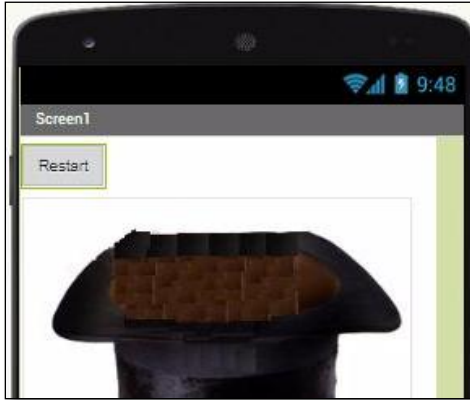


Το κουμπί που βάλαμε στην επιφάνεια εργασίας φαίνεται κάπως έτσι:



Στις ιδιότητές του βρίσκουμε την «Text» και σβήνουμε το κείμενο «Text for button 1» για να μην φαίνεται μέσα στην εικόνα του καπέλου.

Βάζουμε ένα δεύτερο αντικείμενο Button στην κύρια επιφάνεια εργασίας και από τις ιδιότητές του αλλάζουμε το κείμενο από «Text for Button2» σε «Restart»:



Στη στήλη Components υπάρχουν δύο αντικείμενα με παρόμοιο όνομα: Button1, Button2. Αυτά είναι τα «προγραμματιστικά» ονόματα των αντικειμένων, δηλαδή τα ονόματα που συμμετέχουν στον προγραμματισμό. Είναι πολύ βοηθητικό κατά τη διάρκεια του προγραμματισμού να μην χρειάζεται να θυμόμαστε ποιο αντικείμενο είναι το Button1 και ποιο είναι το Button2, γι' αυτό θα τους δώσουμε ονόματα που τα κάνουν να ξεχωρίζουν: HatButton και RestartButton:

Στη στήλη Components επιλέγουμε το αντικείμενο Button1 και κάνουμε κλικ στο κουμπί Rename. Στο ομώνυμο παράθυρο διαλόγου που θα εμφανιστεί βάζουμε ως New name το Hat και πατάμε το κουμπί OK:

Επαναλαμβάνουμε και αλλάζουμε το όνομα του δεύτερου κουμπιού από Button2 σε Reset.

Τώρα στη στήλη των αντικειμένων τα ονόματα φαίνονται έτσι:

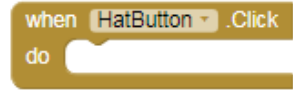
Από την παλέτα επιλέγουμε την ομάδα Media και από αυτήν το αντικείμενο Sound και το τοποθετούμε στην κύρια επιφάνεια εργασίας. Παρατηρούμε ότι θα τοποθετηθεί κάτω από την κύρια επιφάνεια εργασίας και θα εμφανιστεί η ένδειξη Non-visible components:

Προγραμματισμός:

Πατάμε το κουμπί Blocks πάνω δεξιά για να μεταφερθούμε στην οθόνη προγραμματισμού.

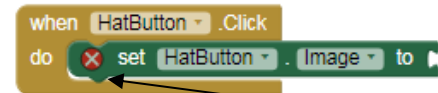
Επιλέγουμε το αντικείμενο HatButton και στη συνέχεια την εντολή

when HatButton.Click do:



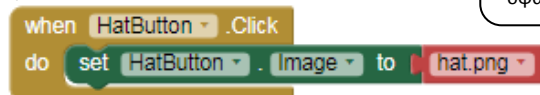
Επιλέγουμε το αντικείμενο HatButton ξανά και από τις εντολές που βρίσκουμε (τραβώντας προς τα κάτω την κατακόρυφη μπάρα κύλισης) την εντολή **set HatButton.Image to** στην προηγούμενη εντολή.

Το μπλοκ εντολών θα μοιάζει κάπως έτσι:



Όταν η σύνταξη μιας εντολής δεν είναι σωστή, εμφανίζεται εικονίδιο σφάλματος.

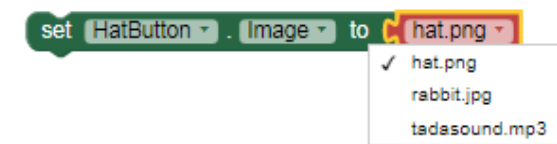
ή έτσι:



Σημείωση:

Η εντολή **set HatButton.Image to** είναι πιθανόν να έχει ήδη «κουμπωμένο» το στοιχείο hat.png (το όνομα του αρχείου του καπέλου). Αν όχι το βρίσκουμε και το κουμπώνουμε εμείς.

Αλλάζουμε από hat.png σε rabbit.jpg όπως φαίνεται παρακάτω:

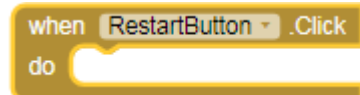


Στο τέλος το μπλοκ εντολών θα μοιάζει έτσι:

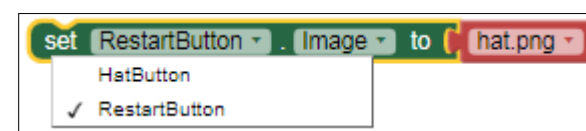


(Οι παραπάνω εντολές σημαίνουν: Όταν πατηθεί το κουμπί με το καπέλο, άλλαξε την εικόνα του σε αυτήν του κουνελιού.)

Επιλέγουμε το αντικείμενο RestartButton και στη συνέχεια την εντολή **when RestatButton.Click do:**



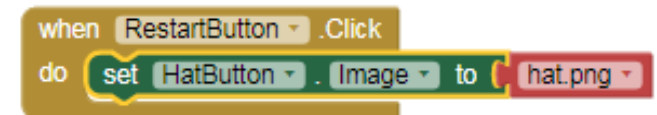
Επιλέγουμε το αντικείμενο RestartButton ξανά και από τις εντολές που βρίσκουμε την εντολή **set HatButton.Image to** και την ρυθμίζουμε:



για να γίνει έτσι:



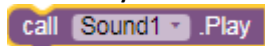
και μετά την κουμπώνουμε στην προηγούμενη εντολή:



(Οι παραπάνω εντολές σημαίνουν: Όταν πατηθεί το κουμπί Restart, άλλαξε την εικόνα του κουμπιού HatButton σε αυτήν με το καπέλο.)

Προσθέτουμε το ηχητικό εφέ στην πρώτη εντολή.

Επιλέγουμε το αντικείμενο Sound1 και στη συνέχεια την εντολή **call Sound1.Play:**



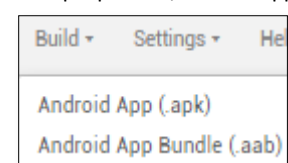
και την κουμπώνουμε στην πρώτη εντολή:



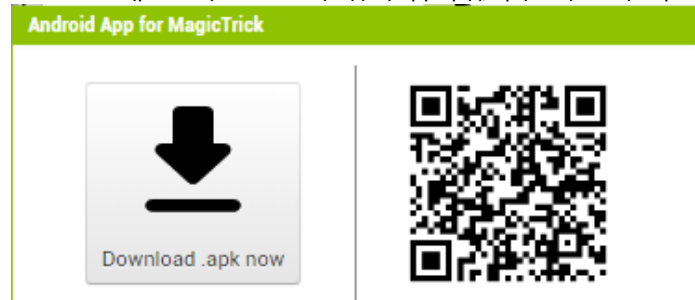
(Οι παραπάνω εντολές σημαίνουν: Όταν πατηθεί το κουμπί με το καπέλο, άλλαξε την εικόνα του σε αυτήν του κουνελιού και παίξε το ηχητικό εφέ.)

Δοκιμάζουμε την εφαρμογή μας:

Επιλέγουμε Build, Android App (.apk)



Όταν ολοκληρωθεί η κατασκευή της εφαρμογής, εμφανίζεται η παρακάτω οθόνη:



Σαρώνουμε με το κινητό μας το QRCode και ακολουθούμε τις οδηγίες για την εγκατάστασή της. Αν εμφανιστούν μηνύματα για επιβλαβές περιεχόμενο, επιλέγουμε ότι η εφαρμογή μας δεν είναι επικίνδυνη και προχωράμε στην εγκατάστασή της για να τη δοκιμάσουμε.

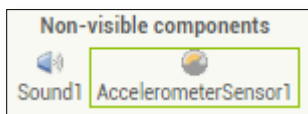
Παραλλαγή της εφαρμογής (χρήση του αισθητήρα επιτάχυνσης του κινητού)

Το κουνέλι θα εμφανίζεται όταν «ταρακουνάμε» το κινητό !

Ας δούμε πώς γίνεται...

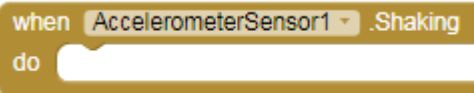
Στην οθόνη σχεδιασμού από την ομάδα Sencors επιλέγουμε το αντικείμενο **AccelerometerSensor** και το τοποθετούμε στην κύρια επιφάνεια εργασίας.

Παρατηρούμε ότι και αυτό το αντικείμενο, όπως και το αντικείμενο Sound, θα τοποθετηθεί κάτω από την κύρια επιφάνεια εργασίας ως μη ορατό στοιχείο:

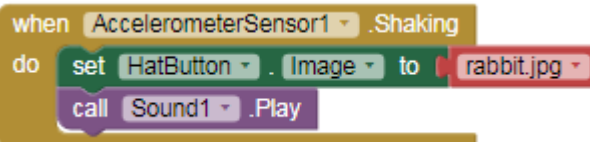


Στην οθόνη προγραμματισμού, κάνουμε τα εξής:

Επιλέγουμε το αντικείμενο **when AccelerometerSensor1.Shaking do :**



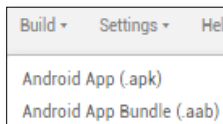
και του κομπώνουμε τις εντολές (που υπάρχουν και στο μπλοκ *when HatButton.Click do*) **set HatButton.Image to rabbit.jpg** και **call Sound1.Play:**



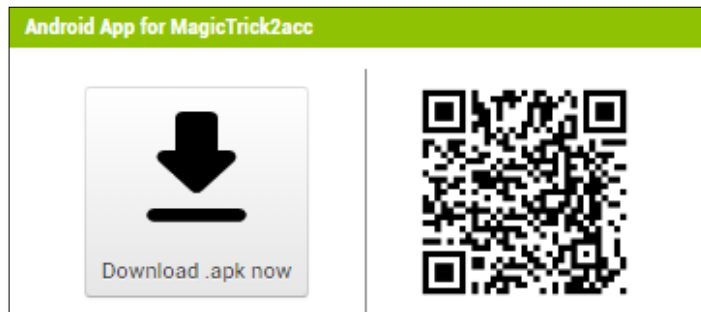
Έτσι κάθε φορά που θα κουνάμε απότομα το κινητό, θα εμφανίζεται το κουνέλι και θα ακούγεται το ηχητικό εφέ.

Δοκιμάζουμε την εφαρμογή μας.

Επιλέγουμε Build, Android App (.apk)



Όταν ολοκληρωθεί η κατασκευή της εφαρμογής, εμφανίζεται η παρακάτω οθόνη:

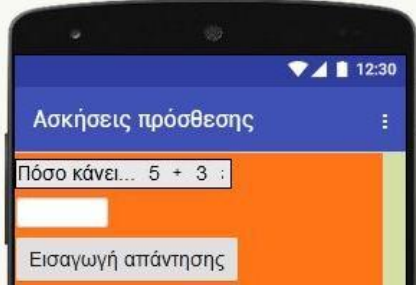


Σαρώνουμε με το κινητό μας το QRCode και ακολουθούμε τις οδηγίες για την εγκατάστασή της. Αν εμφανιστούν μηνύματα για επιβλαβές περιεχόμενο, επιλέγουμε ότι η εφαρμογή μας δεν είναι επικίνδυνη και προχωράμε στην εγκατάστασή της για να τη δοκιμάσουμε.

Τρίτη εφαρμογή: Additions (Ασκήσεις πρόσθεσης)

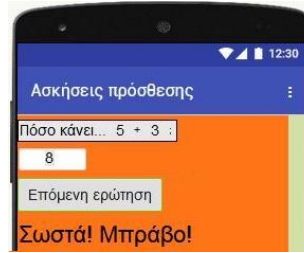
Η εφαρμογή θα επιλέγει δύο τυχαίους από το 1 έως το 10 αριθμούς και θα μας ζητάει να βρούμε το άθροισμά τους.

Η οθόνη της εφαρμογής μοιάζει κάπως έτσι:

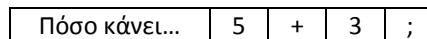


Ο χρήστης θα πληκτρολογεί την απάντησή του στο κενό πλαίσιο και μετά θα πατάει το κουμπί «Εισαγωγή απάντησης».

Θα εμφανίζεται μήνυμα ανατροφοδότησης ανάλογα με το αν είναι σωστή ή λάθος η απάντησή του. Επίσης θα εμφανίζεται ένα νέο κουμπί με τίτλο «Επόμενη ερώτηση», το οποίο θα ξεκινάει τη διαδικασία από την αρχή:



Για να εμφανιστεί η ερώτηση «Πόσο κάνει... 5 + 3 ;» (όπου 5 και 3 τυχαίοι αριθμοί) χρειάζονται 5 αντικείμενα Label:



Σχεδιασμός:

Από την παλέτα επιλέγουμε την ομάδα **User Interface** και από αυτήν το αντικείμενο **Label** και το τοποθετούμε στην κύρια επιφάνεια εργασίας.

Στις ιδιότητές του ρυθμίζουμε τα εξής: Text: «Πόσο κάνει...», FontSize: 20. Αλλάζουμε το προγραμματιστικό όνομά του σε question.

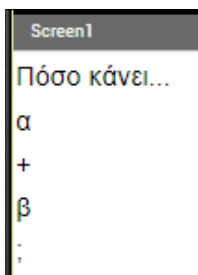
Τοποθετούμε 2^ο αντικείμενο Label. Στις ιδιότητές του ρυθμίζουμε τα εξής: Text: «α», FontSize: 20. Αλλάζουμε το προγραμματιστικό όνομά του σε number1.

Τοποθετούμε 3^ο αντικείμενο Label. Στις ιδιότητές του ρυθμίζουμε τα εξής: Text: «+», FontSize: 20. Αλλάζουμε το προγραμματιστικό όνομά του σε plus.

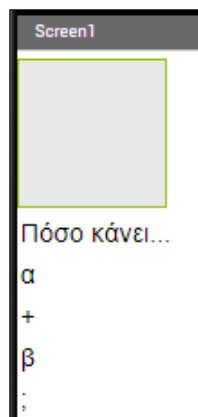
Τοποθετούμε 4^ο αντικείμενο Label. Στις ιδιότητές του ρυθμίζουμε τα εξής: Text: «β», FontSize: 20. Αλλάζουμε το προγραμματιστικό όνομά του σε number2.

Τοποθετούμε 5^ο αντικείμενο Label. Στις ιδιότητές του ρυθμίζουμε τα εξής: Text: «;», FontSize: 20. Αλλάζουμε το προγραμματιστικό όνομά του σε QuestionMark.

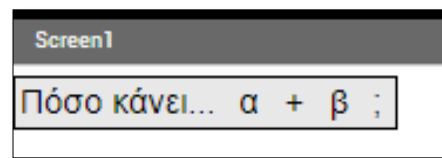
Παρατηρούμε ότι τα αντικείμενα τοποθετήθηκαν το ένα κάτω από το άλλο.



Για να τα βάλουμε σε οριζόντια διάταξη, θα χρειαστούμε ένα αντικείμενο **HorizontalArrangement** από την ομάδα **Layout**. Το βάζουμε πάνω από όλα τα αντικείμενα (εμφανίζεται μια οριζόντια μπλε γραμμή που λειτουργεί ως οδηγός που μας δείχνει πού θα τοποθετηθεί το αντικείμενο.)



Τα σέρνουμε και τα βάζουμε ένα ένα, το ένα δίπλα από το άλλο, μέσα στο πλαίσιο του αντικειμένου **HorizontalArrangement** (το οποίο δείχνει να εξαφανίζεται με το που θα βάλουμε το πρώτο label). (Τώρα εμφανίζεται μια κάθετη μπλε γραμμή που λειτουργεί ως οδηγός που μας δείχνει πού θα τοποθετηθεί το αντικείμενο.)



Το πλαίσιο όπου ο χρήστης πληκτρολογεί την απάντησή του είναι ένα αντικείμενο **TextBox** από την ομάδα User Interface. Το βάζουμε στην κύρια επιφάνεια εργασίας, κάτω από την ερώτηση που υπάρχει ήδη, και ρυθμίζουμε τις ιδιότητές του: Hint: « » (κενό), **FontSize: 20**. Αλλάζουμε το προγραμματιστικό όνομά του σε **answer**.

Το επόμενο αντικείμενο είναι το κουμπί «Εισαγωγή απάντησης».

Αυτό είναι ένα αντικείμενο **Button** από την ομάδα User Interface.

Το βάζουμε στην κύρια επιφάνεια εργασίας και ρυθμίζουμε τις ιδιότητές του:

Text: «Εισαγωγή απάντησης», **FontSize: 20**. Αλλάζουμε το προγραμματιστικό όνομά του σε **AnswerButton**.

Το επόμενο αντικείμενο είναι το κουμπί «Επόμενη ερώτηση». Και αυτό είναι ένα αντικείμενο **Button**.

Το βάζουμε στην κύρια επιφάνεια εργασίας και ρυθμίζουμε τις ιδιότητές του:

Text: «Επόμενη ερώτηση», **FontSize: 20**. Αλλάζουμε το προγραμματιστικό όνομά του σε **NextQButton**

Το επόμενο αντικείμενο είναι το μήνυμα σωστού/λάθους. Αυτό είναι ένα αντικείμενο **Label**.

Το βάζουμε στην κύρια επιφάνεια εργασίας και ρυθμίζουμε τις ιδιότητές του:

Text: «Σωστό/Λάθος» (ενδεικτικό) **FontSize: 30**. Αλλάζουμε το προγραμματιστικό όνομά του σε **RightWrongLabel**.

Τελειώσαμε τον σχεδιασμό! Πάμε για τον προγραμματισμό...

Προγραμματισμός:

Πατάμε το κουμπί Blocks πάνω δεξιά για να μεταφερθούμε στην οθόνη προγραμματισμού.

Αρχικά ορίζουμε δύο μεταβλητές, την **a** και την **b** οι οποίες θα παίρνουν τυχαίες τιμές από το 1 έως το 10.

Από την ομάδα εντολών Variables επιλέγουμε την εντολή **initialize global name to** (αλλάζουμε το name σε a):

Από την ομάδα εντολών Math επιλέγουμε την εντολή **random integer from 1 to 100** (αλλάζουμε το 100 σε 10):

και την κουμπώνουμε στην προηγούμενη. Έτσι τώρα βλέπουμε αυτό:

Επαναλαμβάνουμε τη διαδικασία και για τη μεταβλητή b.

Δεν αρκεί ο αρχικός ορισμός των μεταβλητών.

Οι τιμές τους πρέπει να ορίζονται κάθε φορά που δίνεται νέα ερώτηση.

Για να γίνει αυτό χρειαζόμαστε την εντολή **set to** από την ομάδα εντολών Variables:

Από το χειριστήριό της επιλέγουμε global a και η εντολή γίνεται:

Επαναλαμβάνουμε το ίδιο και για την μεταβλητή b.

Όταν θα ξεκινάει η εφαρμογή θα συμβαίνουν τα εξής:

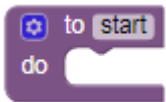
- Η μεταβλητή a θα παίρνει μια τυχαία τιμή από 1 έως 10.
- Η μεταβλητή b θα παίρνει μια τυχαία τιμή από 1 έως 10.
- Στο αντικείμενο number1 (πρώτος αριθμός της ερώτησης) θα εμφανίζεται η τιμή της a.
- Στο αντικείμενο number2 (δεύτερος αριθμός της ερώτησης) θα εμφανίζεται η τιμή της b.
- Το κείμενο στο αντικείμενο RightWrongLabel (μήνυμα σωστού/λάθους) θα σβήνεται (θα μένει κενό).
- Το αντικείμενο AnswerButton (κουμπί «Εισαγωγή απάντησης») θα γίνεται ορατό.
- Το αντικείμενο NextQButton (κουμπί «Επόμενη ερώτηση») θα γίνεται αόρατο.
- Το αντικείμενο answer (πλαίσιο απάντησης) θα αδειάζει.

Τα ίδια ακριβώς θα συμβαίνουν και όταν πατάμε το κουμπί «Επόμενη ερώτηση».

Για να μην ξαναγράψουμε τις ίδιες εντολές δύο φορές, τις γράφουμε μία φορά και τις κάνουμε **διαδικασία (Procedure)**.

Οι διαδικασίες καλούνται από άλλες εντολές για να εκτελεστούν κι αυτή είναι η χρησιμότητά τους (να μην επαναλαμβάνουμε τη συγγραφή εντολών.)

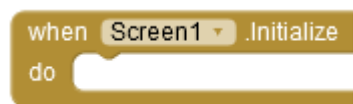
Από την ομάδα εντολών **Procedures** παίρνουμε την εντολή **to procedure do** και αλλάζουμε το όνομα από procedure σε start:



Προσθέτουμε σε αυτήν τις εντολές που κάνουν όσα γράψαμε παραπάνω:

	<p>Η μεταβλητή a θα παίρνει μια τυχαία τιμή από 1 έως 10. Η μεταβλητή b θα παίρνει μια τυχαία τιμή από 1 έως 10. Ο πρώτος αριθμός της ερώτησης θα είναι η τιμή της a. Ο δεύτερος αριθμός της ερώτησης θα είναι η τιμή της b. Το πλαίσιο εισαγωγής της απάντησης αδειάζει. Το μήνυμα σωστού/λάθους σβήνεται (κενό). Το κουμπί «Εισαγωγή απάντησης» γίνεται ορατό. Το κουμπί «Επόμενη ερώτηση» γίνεται αόρατο.</p>
--	--

Βάζουμε τώρα την εντολή που θα ορίζει τι γίνεται όταν ξεκινάει η εφαρμογή.
Είναι η εντολή **when Screen1.Initialize do** από την ομάδα εντολών **Control**:



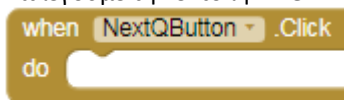
και της κουμπώνουμε την εντολή **call start** από την ομάδα εντολών **Procedures**.

Το μπλοκ εντολών τώρα φαίνεται έτσι:



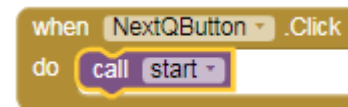
Όταν ξεκινάει η εφαρμογή, θα καλείται (θα εκτελείται) η διαδικασία start (όλες οι εντολές που αυτή περιέχει).

Τα ίδια ακριβώς είπαμε ότι θα συμβαίνουν κι όταν ο χρήστης πατάει το κουμπί «Επόμενη ερώτηση».
Επιλέγουμε την εντολή **when NextQButton.Click do**:

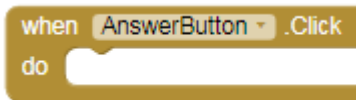


και της κουμπώνουμε την εντολή call start.

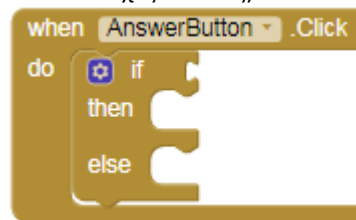
Το μπλοκ εντολών τώρα φαίνεται έτσι:



Το τελευταίο κομμάτι του προγραμματισμού αφορά το κουμπί που ελέγχει την απάντηση του χρήστη.
Επιλέγουμε την εντολή **when AnswerButton.Click do** :



Σε αυτήν κουμπώνουμε την εντολή λογικού ελέγχου **if then else** (από την ομάδα εντολών Control) η οποία εξετάζει αν είναι σωστή ή όχι η απάντηση (και αναλόγως εκτελούνται οι αντίστοιχες εντολές):



Η απάντηση του χρήστη είναι σωστή όταν το περιεχόμενο του πλαισίου answer (αντικείμενο TextBox) είναι ίσο με το άθροισμα των δύο μεταβλητών a και b.

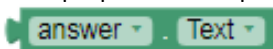
Ας δούμε πώς ελέγχεται αυτό:

Από την ομάδα εντολών Logic επιλέγουμε την εντολή ελέγχου με το ίσον:

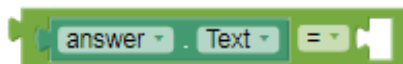


Στην αριστερή μεριά βάζουμε το περιεχόμενο του πλαισίου answer και στη δεξιά το άθροισμα των δύο μεταβλητών:

Επιλέγουμε το αντικείμενο answer και από τις διαθέσιμες εντολές επιλέγουμε την **answer.Text**:



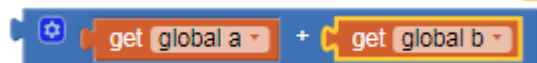
και την κουμπώνουμε στην αριστερή μεριά του ίσον:



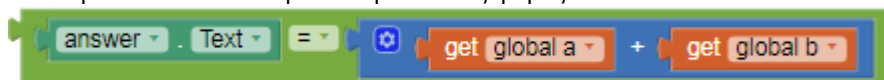
Από την ομάδα εντολών Math επιλέγουμε την εντολή με το συν:



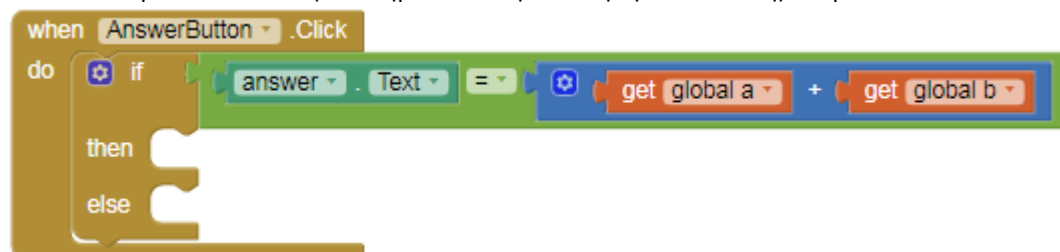
Και μέσα σε αυτήν κουμπώνουμε τις εντολές **get global a** και **get global b** :



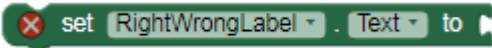
Και το μπλοκ αυτό το κουμπώνουμε στο δεξί μέρος του ίσον:




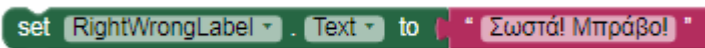
Όλο αυτό το μπλοκ εντολών (που σημαίνει ότι η απάντηση είναι σωστή) κουμπώνει στο if:



Επιλέγουμε το αντικείμενο RightWrongLabel (μήνυμα σωστού/λάθους) και από τις διαθέσιμες εντολές επιλέγουμε την **set RightWrongLabel.Text to** :



Από την ομάδα εντολών Text βρίσκουμε την εντολή με το κενό:  και αφού γράψουμε το μήνυμα «Σωστά! Μπράβο!» την κουμπώνουμε στην προηγούμενη:

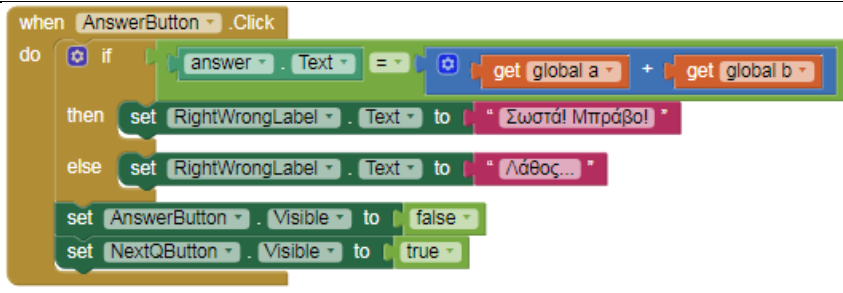


Όλο το παραπάνω μπλοκ κουμπώνει δίπλα στο **then**.

Κάνουμε τις αντίστοιχες ρυθμίσεις και για το **else** (το μήνυμα θα γράφει «Λάθος...»)

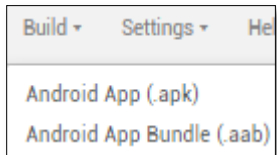
Στη συνέχεια κάνουμε και τις αντίστοιχες ρυθμίσεις για την εμφάνιση ή μη των κουμπιών «Εισαγωγή απάντησης» και «Επόμενη ερώτηση».

Το μπλοκ εντολών στην τελική μορφή του έχει ως εξής:

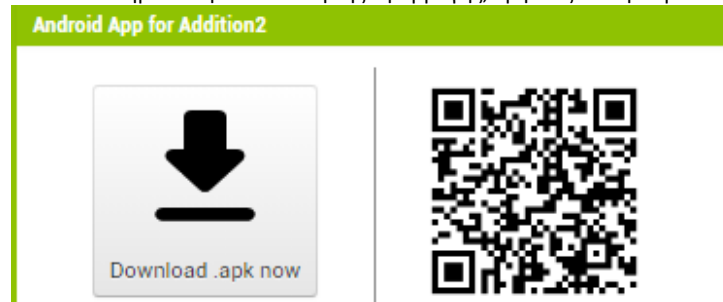
	<p>Όταν πατηθεί το κουμπί «Εισαγωγή απάντησης»</p> <p>Αν η απάντηση είναι σωστή: το μήνυμα σωστού/λάθους θα γράφει «Σωστά! Μπράβο!»</p> <p>αλλιώς (δηλ. αν η απάντηση είναι λάθος) το μήνυμα σωστού/λάθους θα γράφει «Λάθος...»</p> <p>(Άσχετα από αν είναι σωστή ή λάθος η απάντηση) το κουμπί «Εισαγωγή απάντησης» θα γίνεται αόρατο το κουμπί «Επόμενη ερώτηση» θα γίνεται ορατό</p>
--	---

Δοκιμάζουμε την εφαρμογή μας:

Επιλέγουμε Build, Android App (.apk)



Όταν ολοκληρωθεί η κατασκευή της εφαρμογής, εμφανίζεται η παρακάτω οθόνη:



Σαρώνουμε με το κινητό μας το QRCode και ακολουθούμε τις οδηγίες για την εγκατάστασή της. Αν εμφανιστούν μηνύματα για επιβλαβές περιεχόμενο, επιλέγουμε ότι η εφαρμογή μας δεν είναι επικίνδυνη και προχωράμε στην εγκατάστασή της για να τη δοκιμάσουμε.

Τέταρτη εφαρμογή: My Paint App (Ζωγραφική)

Ζωγραφίσουμε με απλές γραμμές με διάφορα χρώματα και σβήνουμε ό,τι έχουμε ζωγραφίσει.
Η οθόνη της εφαρμογής μοιάζει κάπως έτσι:



Σχεδιασμός:

Από την παλέτα επιλέγουμε την ομάδα αντικειμένων Drawing and Animation και από αυτήν επιλέγουμε το αντικείμενο Canvas και το τοποθετούμε στην κύρια επιφάνεια εργασίας. Το ρυθμίζουμε ως εξής:

Height: Fill parent... Width: Fill parent...

Από την ομάδα αντικειμένων Layout επιλέγουμε το αντικείμενο HorizontalArrangement και το τοποθετούμε στην κύρια επιφάνεια εργασίας.

Από την ομάδα αντικειμένων User Interface επιλέγουμε το αντικείμενο Button και το τοποθετούμε στην κύρια επιφάνεια εργασίας, μέσα στο αντικείμενο HorizontalArrangement.

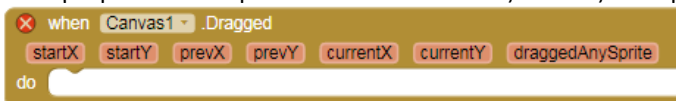
Το ρυθμίζουμε ως εξής: Προγραμματιστικό όνομα: red, BackgroundColor: Red, Height: Fill parent..., Width: 10 percent...

Επαναλαμβάνουμε βάζοντας άλλα τέσσερα αντικείμενα Button για το πράσινο, το μπλε, το κίτρινο και το μαύρο χρώμα, κάνοντας τις αντίστοιχες ρυθμίσεις στα προγραμματιστικά ονόματα και στο χρώμα τους.

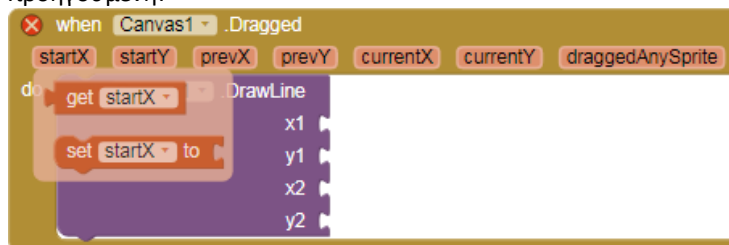
Βάζουμε κι ένα τελευταίο αντικείμενο Button, το οποίο θα έχει προγραμματιστικό όνομα eraser και κείμενο «Σβήστα όλα»

Προγραμματισμός:

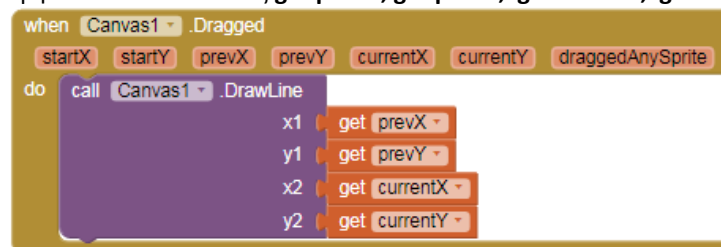
Επιλέγουμε το αντικείμενο Canvas1 και από τις εντολές που εμφανίζονται επιλέγουμε την **when Canvas1.Dragged do**



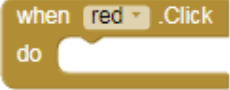

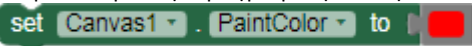
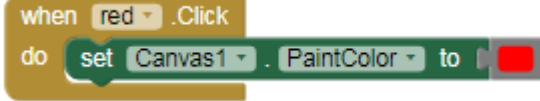
Επιλέγουμε πάλι το αντικείμενο Canvas1 και από τις εντολές επιλέγουμε την **call Canvas1.DrawLine** και την κουμπώνουμε στην προηγούμενη:



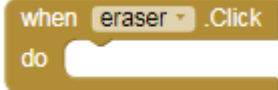
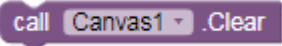
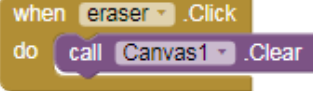
Αφήνουμε τον δείκτη του ποντικιού πάνω στο από το σημείο που γράφει prevX, prevY, current, currentY και μόλις εμφανιστούν οι εντολές **get prevX**, **get prevY**, **get current**, **get currentY** τις σέρνουμε και τις κουμπώνουμε στα x1, y1, x2, y2:



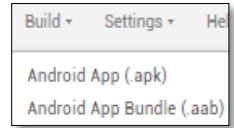
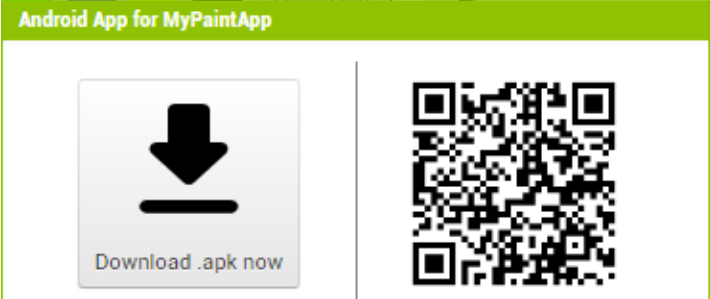
Πώς αλλάζουμε τα χρώματα:

<p>Επιλέγουμε το αντικείμενο red (είναι το κουμπί για το κόκκινο χρώμα) και στη συνέχεια την εντολή when red.Click do</p> 	<p>Επιλέγουμε το αντικείμενο Canvas1 και στη συνέχεια την εντολή set Canvas1.PaintColor to</p>  <p>Από την ομάδα εντολών Colors επιλέγουμε το κόκκινο χρώμα και το κουμπώνουμε στην προηγούμενη εντολή:</p> 
<p>και όλο αυτό στην εντολή του κουμπιού για το κόκκινο χρώμα:</p> 	<p>Επαναλαμβάνουμε τα ίδια για τα υπόλοιπα χρώματα.</p>

Κουμπί που σβήνει ό,τι έχουμε ζωγραφίσει:

<p>Επιλέγουμε το αντικείμενο eraser και στη συνέχεια την εντολή when eraser.Click do</p> 	<p>Επιλέγουμε το αντικείμενο Canvas1 και στη συνέχεια την εντολή call Canvas1.Clear</p>  <p>και την κουμπώνουμε στην προηγούμενη:</p> 
--	---

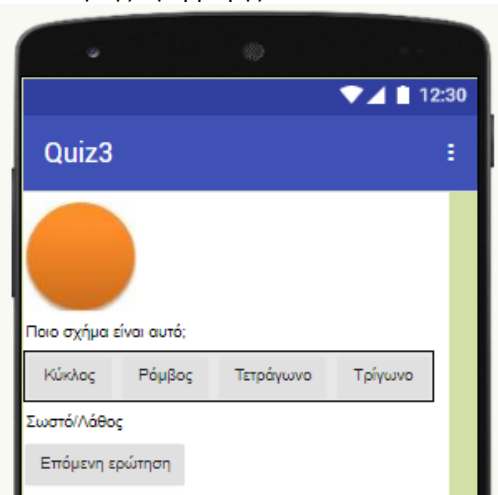
Δοκιμάζουμε την εφαρμογή μας:

<p>Επιλέγουμε Build, Android App (.apk)</p> 	<p>Όταν ολοκληρωθεί η κατασκευή της εφαρμογής, εμφανίζεται η παρακάτω οθόνη:</p>  <p>Σαρώνουμε με το κινητό μας το QRCode και ακολουθούμε τις οδηγίες για την εγκατάστασή της. Αν εμφανιστούν μηνύματα για επιβλαβές περιεχόμενο, επιλέγουμε ότι η εφαρμογή μας δεν είναι επικίνδυνη και προχωράμε στην εγκατάστασή της για να τη δοκιμάσουμε.</p>
--	--

Πέμπτη εφαρμογή: Quiz (Ερωτήσεις με απάντηση πολλαπλής επιλογής)

Η εφαρμογή δείχνει κάποιο γεωμετρικό σχήμα (υπάρχουν τέσσερα βασικά γεωμετρικά σχήματα, τα οποία προβάλλονται με τυχαία σειρά) και ρωτάει το όνομά του, δίνοντας τέσσερις επιλογές απάντησης.

Η οθόνη της εφαρμογής:

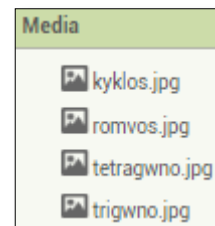


Σχεδιασμός:

Θα βρούμε και θα φορτώσουμε στα Media τις εικόνες τεσσάρων γεωμετρικών σχημάτων (κύκλος, ρόμβος, τετράγωνο, τρίγωνο).

Καλόν είναι να είναι σχετικά μικρά σε διαστάσεις (περίπου 80x80 pixel).

Θα πρέπει να προσέξουμε τα ονόματα των αρχείων να είναι στα αγγλικά.



Τα αντικείμενα που θα χρειαστούν είναι:

Ένα αντικείμενο image (για το γεωμετρικό σχήμα).

Αρχικές ρυθμίσεις: Προγραμματιστικό όνομα: **sxima**, Picture: kyklos.jpg

Ένα αντικείμενο Label (για την ερώτηση).

Αρχικές ρυθμίσεις: Προγραμματιστικό όνομα: **question**, Text: «Ποιο σχήμα είναι αυτό;»

Ένα αντικείμενο **HorizontalArrangement** (για να μπουν οι πιθανές απαντήσεις στην ίδια σειρά).

Δεν χρειάζεται ρυθμίσεις.

Τέσσερα αντικείμενα Button (για τις τέσσερις πιθανές απαντήσεις).

Αρχικές ρυθμίσεις:

Προγραμματιστικό όνομα: **answer1**, Text: «Κύκλος»

Προγραμματιστικό όνομα: **answer2**, Text: «Ρόμβος»

Προγραμματιστικό όνομα: **answer3**, Text: «Τετράγωνο»

Προγραμματιστικό όνομα: **answer4**, Text: «Τρίγωνο»

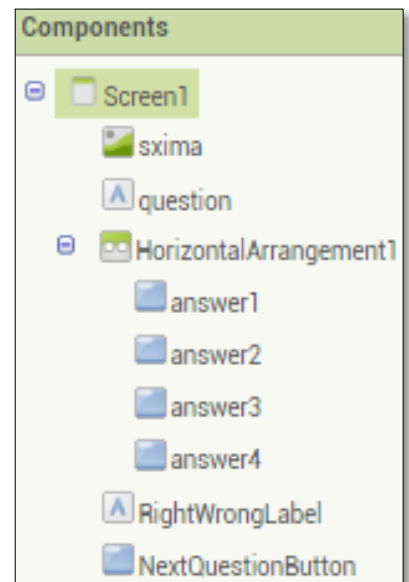
Ένα αντικείμενο Label (για το μήνυμα σωστού/λάθους).

Αρχικές ρυθμίσεις: Προγραμματιστικό όνομα: **RightWrongLabel**, Text: "Σωστό/Λάθος"

(Ενδεικτικό. Οι τελικές τιμές θα είναι «Σωστά! Μπράβο!» για τη σωστή απάντηση και «Λάθος...» για τη λάθος απάντηση.)

Ένα αντικείμενο Button (για το κουμπί «Επόμενη ερώτηση»)

Αρχικές ρυθμίσεις: Προγραμματιστικό όνομα: **NextQuestionButton**, Text: «Επόμενη ερώτηση»



Προγραμματισμός:

Ορίζουμε μια μεταβλητή, την **a** η οποία θα παίρνει τυχαίες τιμές από 1 έως 4 (κι ανάλογα θα εμφανίζεται η αντίστοιχη ερώτηση):



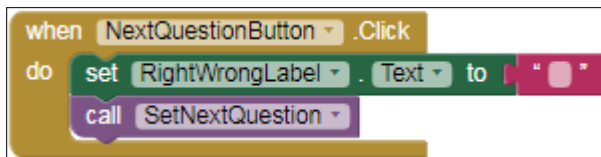
Όταν θα ξεκινάει η εφαρμογή:

<pre>when Screen1.Initialize do set RightWrongLabel.Text to "" call SetNextQuestion</pre>	Το κείμενο του μηνύματος «Σωστό/Λάθος» αδειάζει (μένει κενό) Εκτελείται η διαδικασία εμφάνισης της επόμενης ερώτησης.
---	--

Η διαδικασία (**Procedure**) εμφάνισης της επόμενης ερώτησης (την ονομάζουμε **SetNextQuestion**) κάνει τα εξής:

<pre>to SetNextQuestion do set global a to random integer from 1 to 4 if get global a = 1 then set sxima.Picture to kyklos.jpg else if get global a = 2 then set sxima.Picture to romvos.jpg else if get global a = 3 then set sxima.Picture to tetragwno.jpg else if get global a = 4 then set sxima.Picture to trigwno.jpg</pre>	Η μεταβλητή a παίρνει μια τυχαία τιμή από 1 έως 4 Εάν η a είναι 1 τότε το αντικείμενο sxima γίνεται κύκλος Αλλιώς, εάν η a είναι 2 τότε το αντικείμενο sxima γίνεται ρόμβος Αλλιώς, εάν η a είναι 3 τότε το αντικείμενο sxima γίνεται τετράγωνο Αλλιώς, εάν η a είναι 4 τότε το αντικείμενο sxima γίνεται τρίγωνο
--	--

Ό,τι γίνεται όταν ξεκινάει η εφαρμογή, θα γίνεται και όταν πατιέται το κουμπί «Επόμενη ερώτηση»:

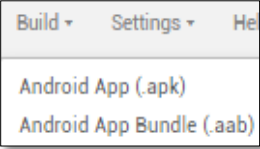



Τι γίνεται όταν πατιούνται τα κουμπιά των τεσσάρων πιθανών απαντήσεων:

<p>Όταν πατιέται το κουμπί answer1, τότε:</p> <pre>when answer1.Click do if sxima.Picture = kyklos.jpg then set RightWrongLabel.Text to "Σωστά! Μπράβο!" else set RightWrongLabel.Text to "Λάθος..."</pre>	Εάν το αντικείμενο sxima είναι κύκλος Τότε εμφάνισε το μήνυμα «Σωστά! Μπράβο!» Αλλιώς εμφάνισε το μήνυμα «Λάθος...»
<p>Όταν πατιέται το κουμπί answer2, τότε:</p> <pre>when answer2.Click do if sxima.Picture = romvos.jpg then set RightWrongLabel.Text to "Σωστά! Μπράβο!" else set RightWrongLabel.Text to "Λάθος..."</pre>	Εάν το αντικείμενο sxima είναι ρόμβος Τότε εμφάνισε το μήνυμα «Σωστά! Μπράβο!» Αλλιώς εμφάνισε το μήνυμα «Λάθος...»

<p>Όταν πατιέται το κουμπί answer3, τότε:</p> <pre> when answer3 .Click do if sxima . Picture = tetragwno.jpg then set RightWrongLabel . Text to " Σωστά! Μπράβο! " else set RightWrongLabel . Text to " Λάθος... " </pre>	<p>Εάν το αντικείμενο σχημα είναι τετράγωνο Τότε εμφάνισε το μήνυμα «Σωστά! Μπράβο!» Αλλιώς εμφάνισε το μήνυμα «Λάθος...»</p>
<p>Όταν πατιέται το κουμπί answer4, τότε:</p> <pre> when answer4 .Click do if sxima . Picture = trigwno.jpg then set RightWrongLabel . Text to " Σωστά! Μπράβο! " else set RightWrongLabel . Text to " Λάθος... " </pre>	<p>Εάν το αντικείμενο σχημα είναι τρίγωνο Τότε εμφάνισε το μήνυμα «Σωστά! Μπράβο!» Αλλιώς εμφάνισε το μήνυμα «Λάθος...»</p>

Δοκιμάζουμε την εφαρμογή μας:

<p>Επιλέγουμε Build, Android App (.apk)</p> 	<p>Όταν ολοκληρωθεί η κατασκευή της εφαρμογής, εμφανίζεται η παρακάτω οθόνη:</p>  <p>Σαρώνουμε με το κινητό μας το QRCode και ακολουθούμε τις οδηγίες για την εγκατάστασή της. Αν εμφανιστούν μηνύματα για επιβλαβές περιεχόμενο, επιλέγουμε ότι η εφαρμογή μας δεν είναι επικίνδυνη και προχωράμε στην εγκατάστασή της για να τη δοκιμάσουμε.</p>
--	---

Έκτη εφαρμογή: QuizGG7 (Γεωγραφία Ελλάδας)

Η εφαρμογή περιέχει επτά ερωτήσεις με 4 πιθανές απαντήσεις η καθεμία, όλες σε μία οθόνη (κυλιόμενη).

Κάθε απάντηση που επιλέγει ο χρήστης πρασινίζει (προσωρινά).

Αφού τις απαντήσει όλες, πατάει το κουμπί «Έλεγχος» και τότε:

Όποια απάντηση είναι σωστή, παραμένει πράσινη.

Όποια απάντηση είναι λάθος γίνεται κόκκινη και ταυτόχρονα επισημαίνεται με πράσινο χρώμα η σωστή.

Η οθόνη της εφαρμογής:

The image shows two side-by-side panels. The left panel is a screenshot of the QuizGG7 app interface. The title bar at the top is blue and says "Γεωγραφία Ελλάδας". Below it, there are two buttons: "Τριχωνίδα" (highlighted in green) and "Υλίκη" (grey). The first question is "Ποιο είναι το ψηλότερο βουνό της Ελλάδας;". Below it are four buttons: "Όσσα" (grey), "Όλυμπος" (highlighted in green), "Ταίναρο" (grey), and "Παρνασσός" (grey). The second question is "Ποια είναι η μεγαλύτερη πεδιάδα της Ελλάδας;". Below it are four buttons: "Θεσσαλίας" (highlighted in red), "Ξάνθης" (grey), "Σερρών" (grey), and "Κεντρικής Μακεδονίας" (highlighted in green). The third question is "Ποιο είναι το μεγαλύτερο νησί της Ελλάδας;". Below it are four buttons: "Κρήτη" (highlighted in green), "Εύβοια" (grey), "Ρόδος" (grey), and "Κέρκυρα" (grey). The fourth question is "Ποιος είναι ο μεγαλύτερος σε έκταση νομός της Ελλάδας;". Below it are four buttons: "Λάρισας" (grey), "Αττικής" (highlighted in red), "Αιτωλοακαρνανίας" (highlighted in green), and "Ιωαννίνων" (grey). The fifth question is "Ποιος είναι ο πιο αραιοκατοικημένος νομός της Ελλάδας;". Below it are four buttons: "Αρκαδίας" (grey), "Γρεβενών" (highlighted in green), "Φωκίδας" (grey), and "Ευρυτανίας" (highlighted in red). At the bottom of the app screen, it says "Βρήκες 3 στα 7." and "Επανάληψη". The right panel is titled "QuizGG7, QR Code" and has a green header "Android App for Quiz_GG7". It contains a button with a downward arrow and the text "Download .apk now" and a QR code.